

FIG. 1

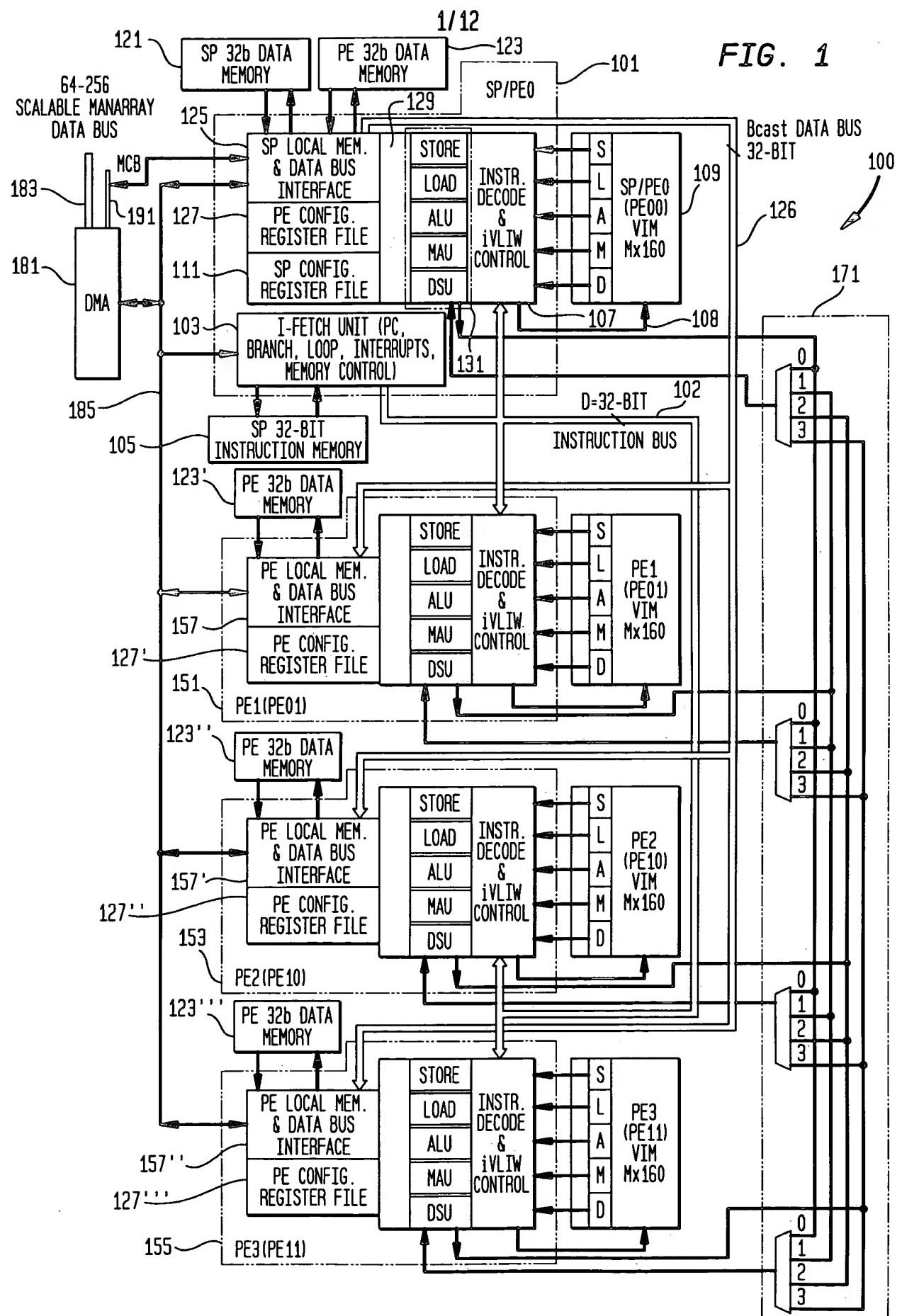


FIG. 2A

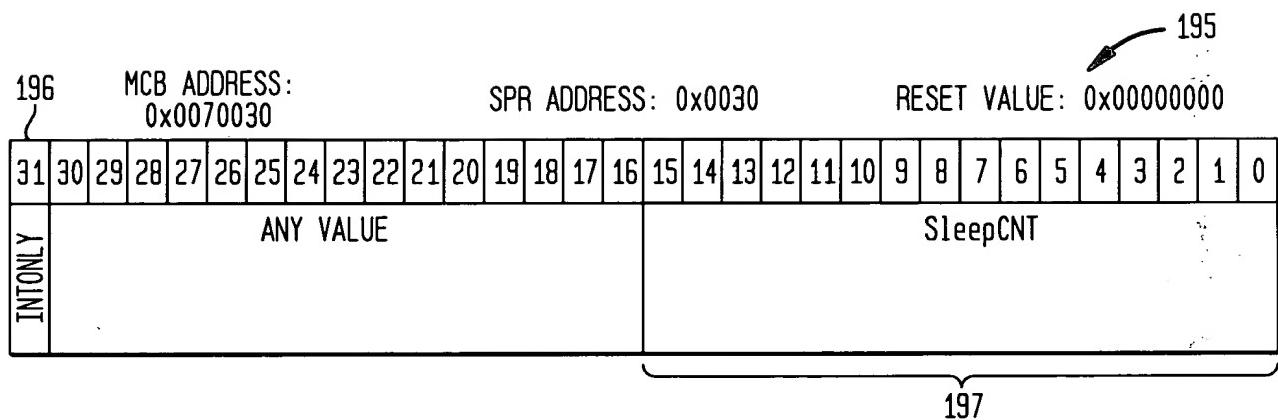


FIG. 2B

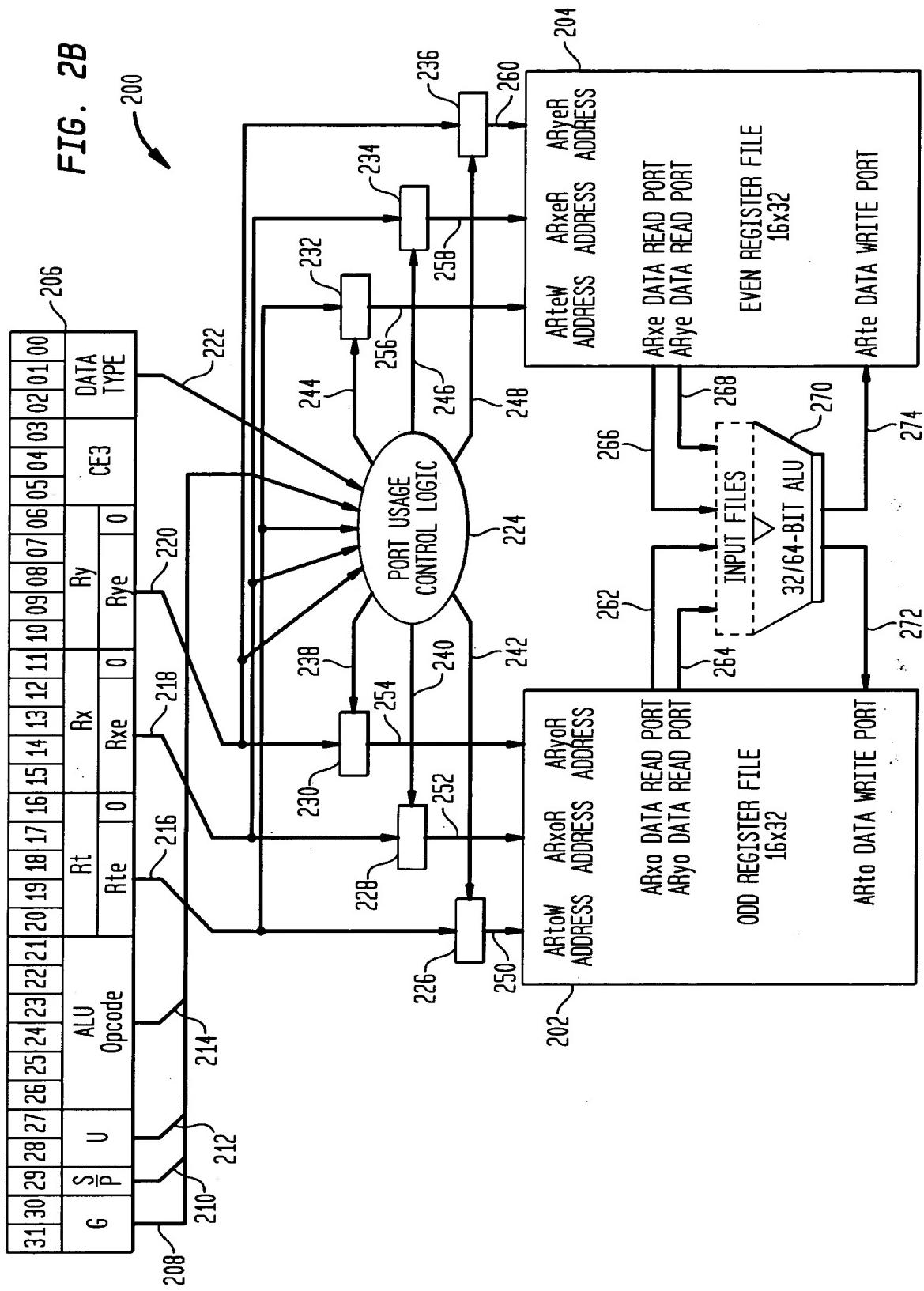


FIG. 3A

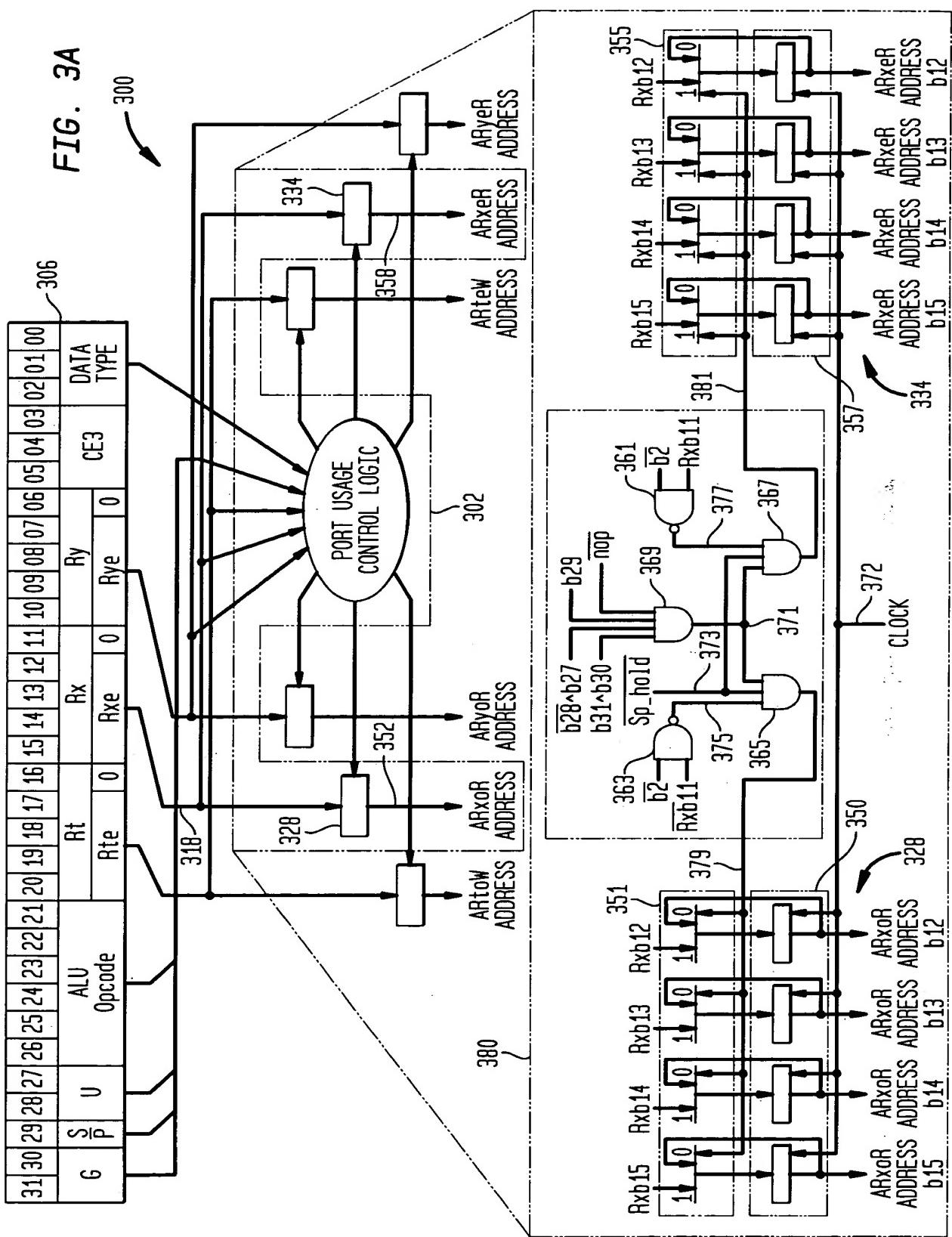


FIG. 3B

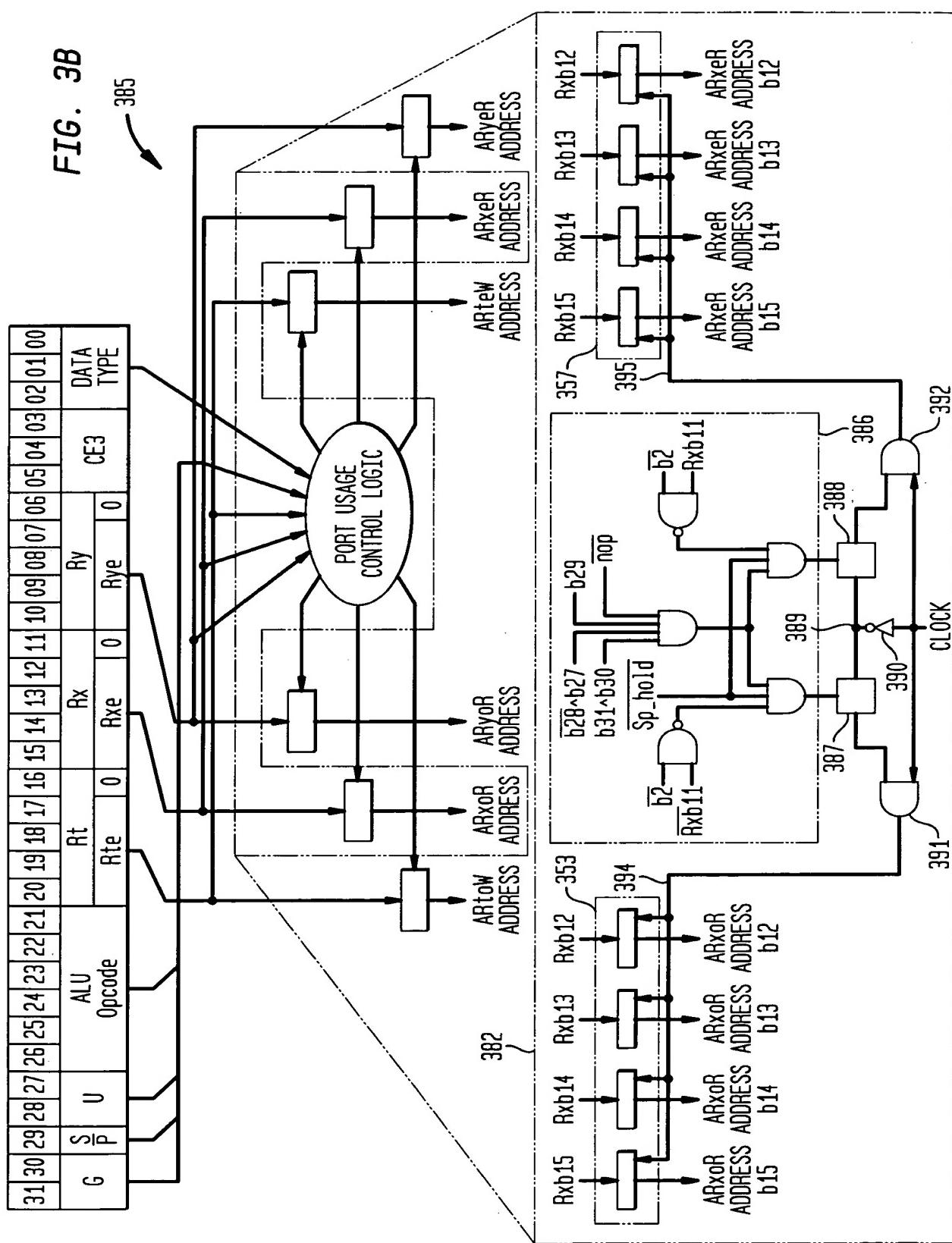
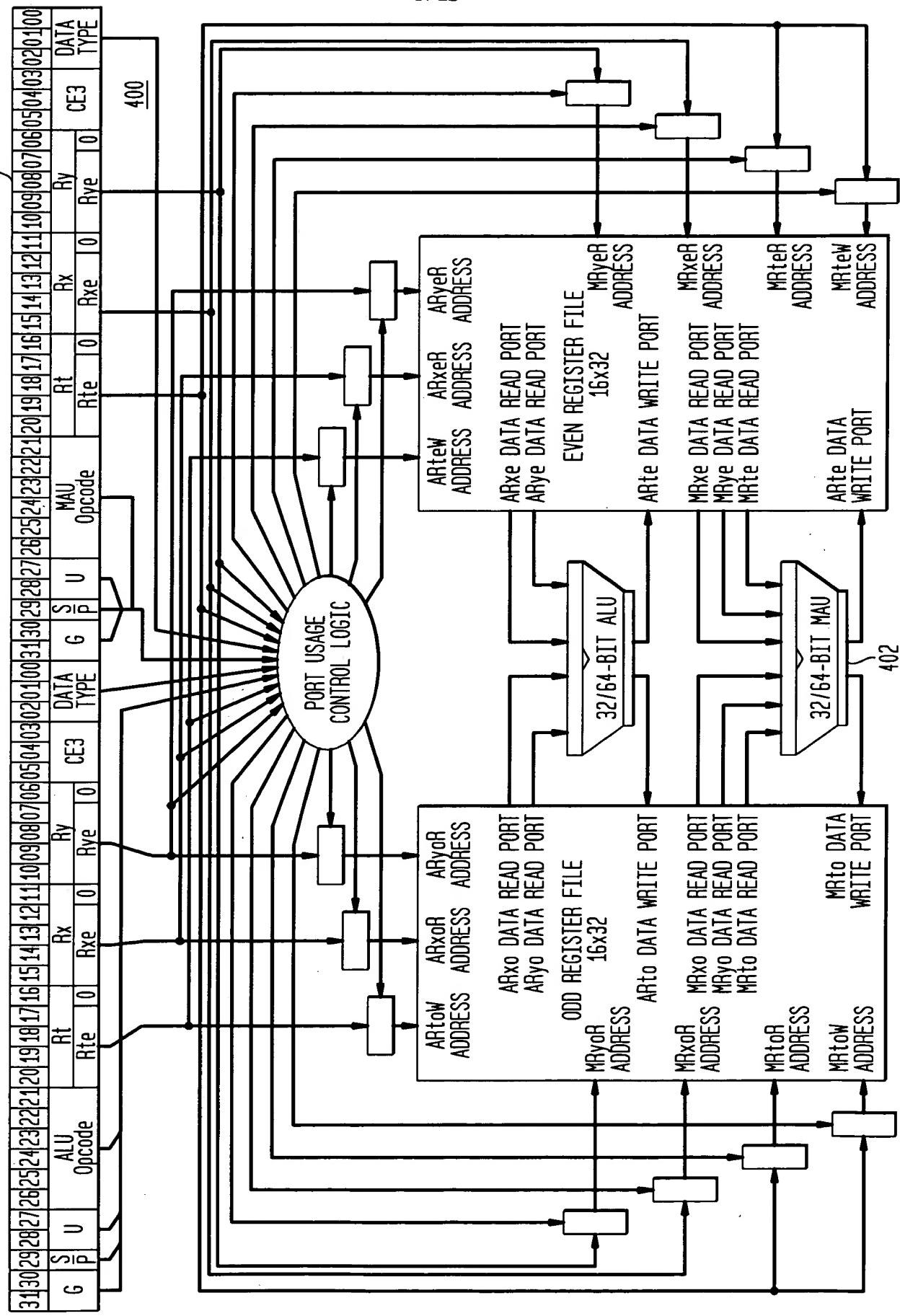
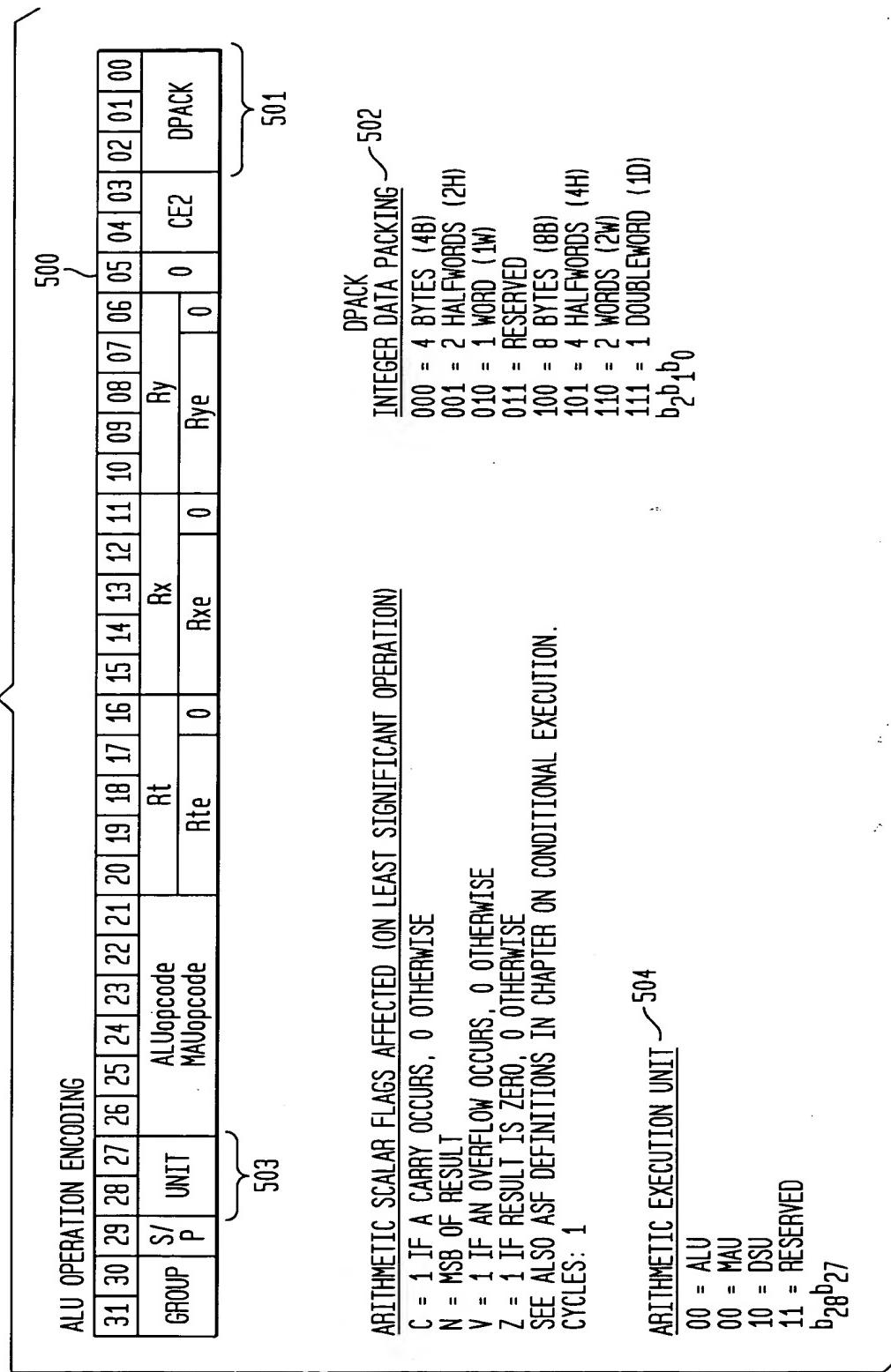


FIG. 4



7/12

FIG. 5A



8/12
FIG. 5B

DESCRIPTION

THE SUM OF SOURCE REGISTERS Rx AND Ry IS STORED IN TARGET REGISTER Rt.
SYNTAX/OPERATION

520

INSTRUCTION	OPERANDS	OPERATION	ACF
DOUBLEWORD			
ADD.[SP][AM].1D	Rte, Rx _e , Ry _e	Rt _o Rt _e ← Rx _o Rx _e + Ry _o Ry _e	NONE
[TF].ADD.[SP][AM].1D	Rte, Rx _e , Ry _e	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE
WORD			
ADD.[SP][AM].1W	Rt, Rx, Ry	Rt ← Rx + Ry	NONE
[TF].ADD.[SP][AM].1W	Rt, Rx, Ry	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE
DUAL WORDS			
ADD.[SP][AM].2W	Rte, Rx _e , Ry _e	Rt _o ← Rx _o + Ry _o Rt _e ← Rx _e + Ry _e	NONE
[TF].ADD.[SP][AM].2W	Rte, Rx _e , Ry _e	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE
DUAL HALFWORDS			
ADD.[SP][AM].2H	Rt, Rx, Ry	Rt.H1 ← Rx.H1 + Ry.H1 Rt.H0 ← Rx.H0 + Ry.H0	NONE
[TF].ADD.[SP][AM].2H	Rt, Rx, Ry	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE
QUAD HALFWORDS			
ADD.[SP][AM].4H	Rte, Rx _e , Ry _e	Rt _o .H1 ← Rx _o .H1 + Ry _o .H1 Rt _o .H0 ← Rx _o .H0 + Ry _o .H0 Rt _e .H1 ← Rx _e .H1 + Ry _e .H1 Rt _e .H0 ← Rx _e .H0 + Ry _e .H0	NONE
[TF].ADD.[SP][AM].4H	Rte, Rx _e , Ry _e	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE
QUAD BYTES			
ADD.[SP][AM].4B	Rt, Rx, Ry	Rt.B3 ← Rx.B3 + Ry.B3 Rt.B2 ← Rx.B2 + Ry.B2 Rt.B1 ← Rx.B1 + Ry.B1 Rt.B0 ← Rx.B0 + Ry.B0	NONE
[TF].ADD.[SP][AM].4B	Rt, Rx, Ry	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE
OCTAL BYTES			
ADD.[SP][AM].8B	Rte, Rx _e , Ry _e	Rt _o .B3 ← Rx _o .B3 + Ry _o .B3 Rt _o .B2 ← Rx _o .B2 + Ry _o .B2 Rt _o .B1 ← Rx _o .B1 + Ry _o .B1 Rt _o .B0 ← Rx _o .B0 + Ry _o .B0 Rt _e .B3 ← Rx _e .B3 + Ry _e .B3 Rt _e .B2 ← Rx _e .B2 + Ry _e .B2 Rt _e .B1 ← Rx _e .B1 + Ry _e .B1 Rt _e .B0 ← Rx _e .B0 + Ry _e .B0	NONE
[TF].ADD.[SP][AM].8B	Rte, Rx _e , Ry _e	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE

9/12

MPYA-MULTIPLY ACCUMULATE
ENCODING

FIG. 6A

600

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
GROUP	S/P	UNIT	MAUopcode		Rte	0	Rx	Ry		CE3	MPACK																				

SYNTAX/OPERATION

FIG. 6B

620

INSTRUCTION	OPERANDS	OPERATION	ACF
WORD			
MPYA.[SP]M.1[SU]W	Rte, Rx, Ry	DO OPERATION BELOW BUT DO NOT AFFECT ACFs	NONE
MPYA[CNVZ].[SP]M.1[SU]W	Rte, Rx, Ry	Rto Rte \leftarrow Rto Rte + (Rx * Ry)	F0
[TF].MPYA.[SP]M.1[SU]W	Rte, Rx, Ry	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN ACFs	NONE
DUAL HALFWORDS			
MPYA.[SP]M.2[SU]H	Rte, Rx, Ry	DO OPERATION BELOW BUT DO NOT AFFECT ACFs	NONE
MPYA[CNVZ].[SP]M.2[SU]H	Rte, Rx, Ry	Rto \leftarrow Rto + (Rx.H1 * Ry.H1) Rte \leftarrow Rte + (Rx.H0 * Ry.H0)	F1 F0
[TF].MPYA.[SP]M.2[SU]H	Rte, Rx, Ry	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN ACFs	NONE
QUAD BYTES			
MPYA.[SP]M.4[SU]B	Rte, Rx, Ry	DO OPERATION BELOW BUT DO NOT AFFECT ACFs	NONE
MPYA[CNVZ].[SP]M.4[SU]B	Rte, Rx, Ry	Rto.H1 \leftarrow Rto.H1 + (Rx.B3 * Ry.B3) Rto.H0 \leftarrow Rto.H0 + (Rx.B2 * Ry.B2) Rte.H1 \leftarrow Rte.H1 + (Rx.B1 * Ry.B1) Rte.H0 \leftarrow Rte.H0 + (Rx.B0 * Ry.B0)	F3 F2 F1 F0
[TF].MPYA.[SP]M.4[SU]B	Rte, Rx, Ry	DO OPERATION ONLY IF T/F CONDITION IS SATISFIED IN F0	NONE

FIG. 6C

640

ARITHMETIC SCALAR FLAGS AFFECTED
(ON LEAST SIGNIFICANT OPERATION)

C = NOT AFFECTED

N = MSB OF RESULT

V = NOT AFFECTED

Z = 1 IF RESULT IS ZERO,
0 OTHERWISE

CYCLES: 2

ARITHMETIC EXECUTION UNIT

00 = ALU

00 = MAU

10 = DSU

11 = RESERVED

 $b_{28}b_{27}$ INSTRUCTION GROUP

00 = RESERVED

00 = FLOW CONTROL

10 = LOAD/STORE (LU, SU)

11 = ARITHMETIC/LOGICAL
(ALU, MAU, DSU) $b_{31}b_{30}$ MPACK-MULTIPLY DATA PACKING

000 = RESERVED

001 = 2 HALFWORDS (2H)

010 = 1 WORD (1W)

011 = RESERVED

100 = RESERVED

101 = 4 HALFWORDS (4H)
FOR MPYH AND MPYL

110 = RESERVED

111 = RESERVED

 $b_2b_1b_0$ SP/PE SELECT

0 = SP

1 = PE

FIG. 7A

RECEIVED 32-BIT XV INSTRUCTION

ENABLE MASK BITS

31 30	29 28	25 24	23 22	21 20	19 18	17	10 9	8	7
Group S/P	XV opcode	Vx	UAF	RFI	0 0	0 0	S L A M D	Vb 0	VIMOFFS

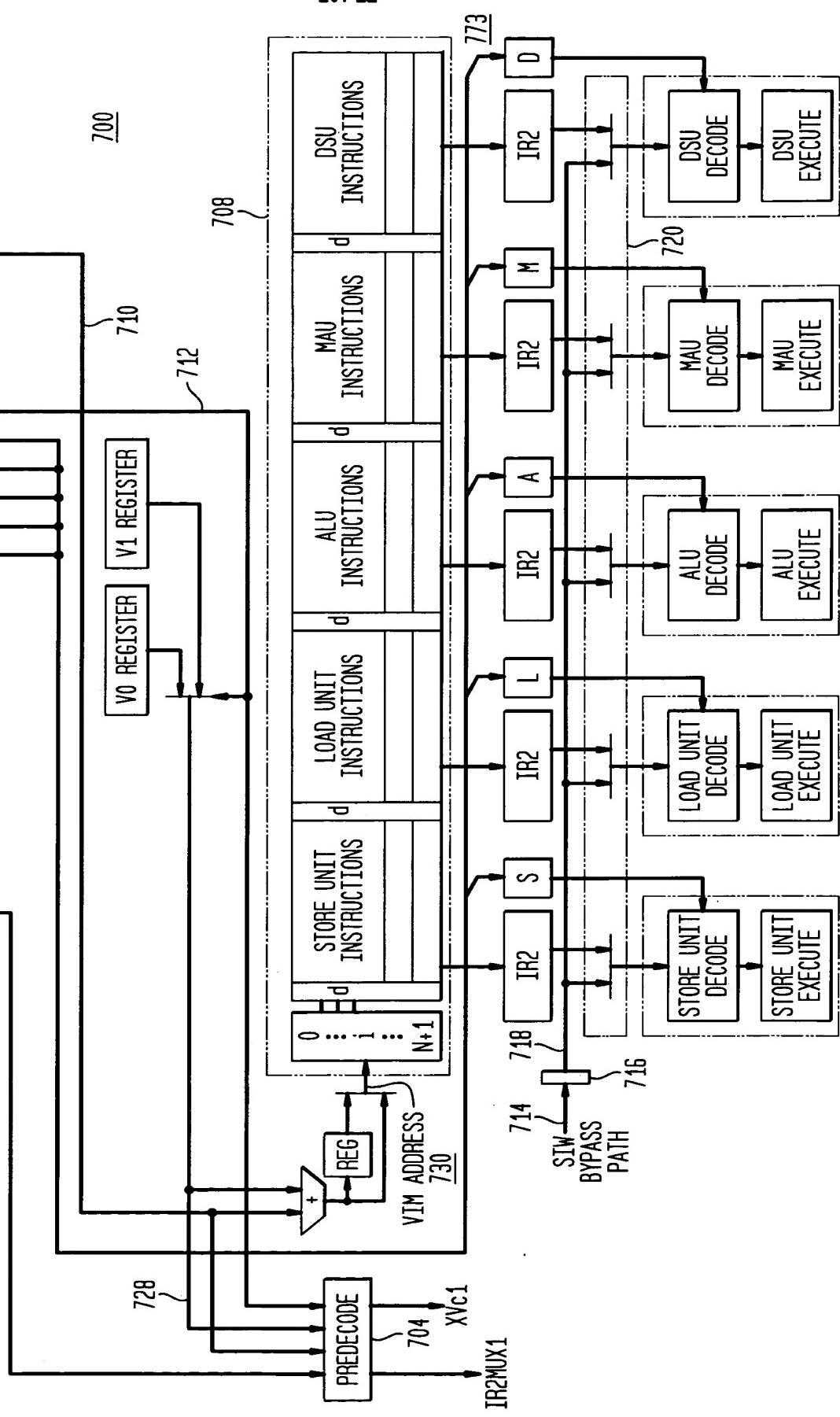


FIG. 7B

RECEIVED 32-BIT XV INSTRUCTION

ENABLE MASK BITS

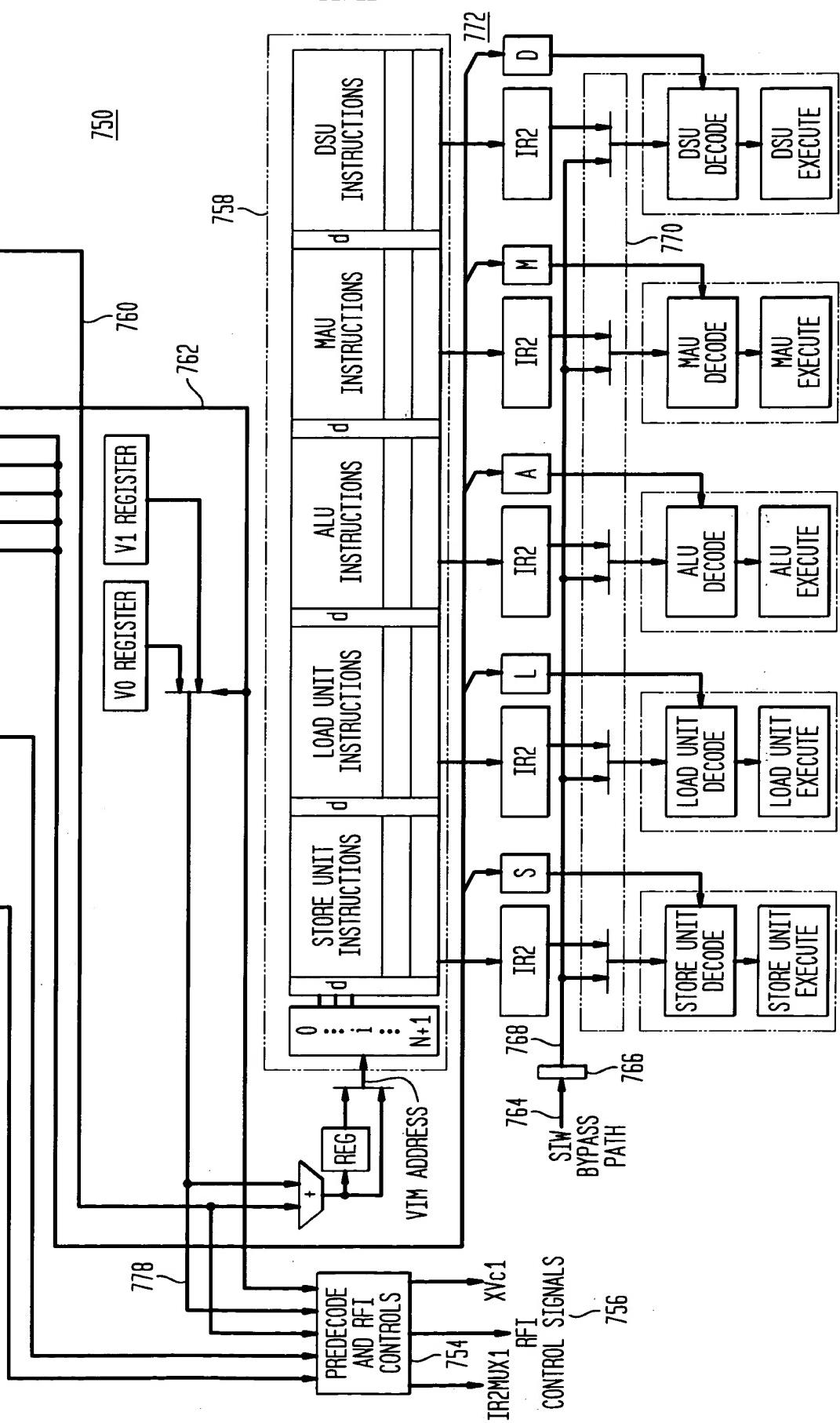
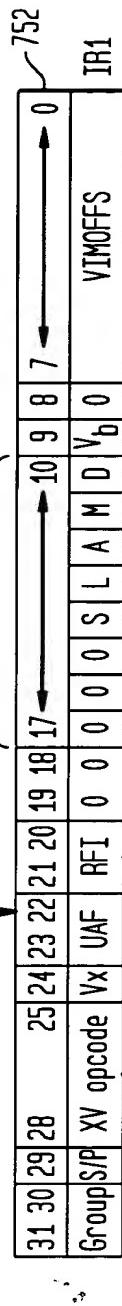


FIG. 8

